

# Usability- Testing App Concepts

ONCE YOU'VE PROTOTYPED your app concepts, you may reach a crossroads:

Should you conduct usability tests now or wait until you have a beta?<sup>1</sup>

Developers tend to take the beta route, but you may save time and money by usability-testing your prototype *before* writing code.

This chapter starts with an overview of usability testing and its benefits.

We'll look at a variety of usability-testing methods—ranging from

“traditional” tests to the RITE method and guerrilla testing—and suggest

how to choose the right approach for your app. We'll also discuss beta testing and ways to enhance it with usability methods.

This chapter also includes a case study on REALTOR.com's iPhone app. Here

we learn how the REALTOR.com team incorporated paper prototyping and

usability testing into the app design process.

---

1. Craig Hockenberry, “Beta Testing on iPhone 2.0,” <http://furbo.org/2008/08/06/beta-testing-on-iphone-20/> (August 2008).

## What Is Usability Testing?

### NOTE

The terms *usability testing* and *user testing* are often used interchangeably. User testing may be less preferable since it suggests that users are being tested, when in fact the design is being tested.

*Usability testing* is an umbrella term for a variety of methods that involve observing users as they interact with a product, typically with a set of predefined tasks (FIGURE 8.1). In addition to evaluating traditional usability metrics—learnability, efficiency, memorability—the sessions can address other user experience concerns. For example, to better understand whether an app meets user needs, you might want to start with semi-structured interviews, as described in Chapter 3, “Introduction to User Research.”



**FIGURE 8.1** Facilitator (right) showing a participant (left) a paper prototype for REALTOR.com  
(Courtesy of Shohini Solanski, photographer)

## Why Usability Testing?

Running usability tests before coding your app can save valuable time and money. If you discover critical user experience issues, it often costs less to change your prototype than to change a fully coded application. Cost savings may also be seen in terms of customer service—fewer user experience issues may mean fewer customer support requests. Perhaps the biggest benefit is increased customer satisfaction. Other important reasons for usability testing are discussed in this section.

### HELP RESOLVE KNOWN DESIGN ISSUES

Over the course of designing your app, you may encounter design problems that could benefit from user feedback. These design problems can be low-level (e.g., information layout) or high-level (e.g., user flows). For example, imagine that your

team has been exploring two different “Getting Started” flows and is uncertain which one would be more effective. You could create two paper prototypes and evaluate both of them with users.

## UNCOVER UNKNOWN DESIGN ISSUES

After several design iterations, you may have a nagging feeling that you missed some critical user experience issues. Even the most skilled designers readily admit that it’s challenging to uncover all design problems solely through heuristic evaluations or self-critiques. Of course, you can supplement them with internal design reviews, but your colleagues may be too familiar with the app, and thus it will be difficult for them to objectively evaluate it. Usability testing, on the other hand, is an effective way to objectively uncover a wide range of unknown design issues.

## SET A BASELINE FOR FUTURE STUDY

Usability testing can be used to establish a baseline for future studies. For example, let’s say you conduct a study and discover that only four out of ten participants could discover your app’s sharing feature. After redesigning the sharing experience, if eight out of ten participants can now find the sharing feature, you may report that sharing discovery doubled. That being said, be careful when citing causal relationships—your argument will not hold up if your study variables were poorly controlled.

## GATHER INFORMATION FOR THE NEXT RELEASE

Once your app is in the App Store, you will receive user feedback through App Store reviews, customer support, and potentially the blogosphere. While these channels are valuable, they may represent a small fraction of your user base. Moreover, the feedback from these channels tends to fall within the realm of “I want feature X” or “I don’t like the latest update.” This information is valuable, but it usually doesn’t tell the whole story behind feature requests and frustrations. Running user tests with a representative cross section of your user base can provide insights into user feedback and guide design improvements for your next release. Keep in mind that it’s better to gather this information *before* launching your product.

## GET STAKEHOLDER BUY-IN

Companies sometimes run user tests to get stakeholder buy-in for a particular feature or design direction. Although this is a relatively common reason, it’s not always a particularly good one. If your colleagues or company executives are questioning your design decisions, they may continue to do so even after usability

testing. Before diving into usability testing, I recommend trying to understand the rationale behind stakeholder concerns. If you can address these concerns internally, it's far better than wasting your users' time to resolve company battles.

### What If I Don't Have Usability-Testing Experience?

This chapter aims at providing you with the tools and guidance needed to run your own usability studies. If you don't feel prepared after reading this chapter, you may want to look into additional training.

The Usability Professionals' Association (UPA) regularly holds conferences around the world which include a range of practical usability workshops and tutorials. Also, consider reading Jeffrey Rubin's excellent book on usability testing: *Handbook of Usability Testing, Second Edition*, published by Wiley in 2008.

## Role of Context

As mentioned in Chapter 3, studies exploring the importance of context in mobile research have produced inconsistent results. Some suggest that field-based studies have more "ecological validity" (the methods, materials, and setting approximate the real-life situation under investigation) and uncover a larger number of critical usability issues,<sup>2</sup> whereas other studies contend that there's little difference between field- and lab-based studies.<sup>3</sup>

In deciding whether to run your tests in the lab or the field, consider the app and the study goals. If you're creating an app where context is a defining aspect of the user experience (e.g., with a location-based app), your tests will ideally take place in the field. However, if you're trying to resolve early-stage flow and terminology issues, you may be able to simulate the context in a lab. In this case you could start with a lab-based paper prototype, then conduct field-based studies in the later design stages. If you are creating an app where context is *not* a defining aspect of the user experience (e.g., with certain stand-alone games), it may suffice to conduct all of your studies in the lab. In short, the field is ideal, but if it's not possible, consider how context affects your app and adapt your approach as needed.

---

2. Christian Monrad Nielsen et al., "It's Worth the Hassle! The Added Value of Evaluating the Usability of Mobile Systems in the Field," *Proceedings of the 4th Nordic Conference on Human-Computer Interaction: Changing Roles* (ACM, 2006), [www.usabilityprofessionals.org/upa\\_publications/jus/2005\\_november/mobile.html](http://www.usabilityprofessionals.org/upa_publications/jus/2005_november/mobile.html).

3. Anne Kaikkonen et al., "Usability Testing of Mobile Applications: A Comparison between Laboratory and Field Testing," *Journal of Usability Studies* (November 2005).

# Usability-Testing Methods

In this section we'll introduce three usability-testing methods: "traditional" usability testing, the RITE method, and paper prototype testing. Later in the chapter we'll delve into the nuts and bolts of running and analyzing sessions.

## TRADITIONAL USABILITY TESTING

"Traditional" usability testing is perhaps the most commonly used method. In essence, it involves observing users one by one as they use your product to complete tasks. While they work through these tasks, the participants are asked to "think aloud," which helps the moderator understand the reasons behind the participants' behavior. For example, knowing that six out of eight participants couldn't find a particular button is not as helpful as knowing *why* they couldn't find the button:

- Was it the label?
- The placement?
- The underlying concept?

## THE RITE METHOD

The Rapid Iterative Testing and Evaluation method—or RITE method—was coined and authored by a team of designers and researchers at Microsoft. The RITE method has many similarities to "traditional" usability testing: Study participants complete tasks and think out loud. The key difference is that RITE emphasizes rapid changes and verification of the effectiveness of these changes.<sup>4</sup> Instead of the designs being revised at the end of the study, the designs are improved after each participant. Given that the time required to fix problems can vary, the authors of RITE created rules for categorizing issues:

**Category 1:** Issues that appear to have an obvious cause and an obvious solution that can be implemented quickly (e.g., text changes, relabeling buttons, rewording overlay text)

**Category 2:** Issues that appear to have an obvious cause and an obvious solution that cannot be implemented quickly or within the time frame of the current test (e.g., difficult new features, current features that require substantial design and code changes)

---

4. M. C. Medlock, D. Wixon, M. Terrano, R. Romero, and B. Fulton, "Using the RITE Method to Improve Products: A Definition and a Case Study," [www.microsoft.com/downloads/details.aspx?FamilyID=3b882eb1-5f06-41d9-baba-d39ad13bc3ff&displaylang=en/](http://www.microsoft.com/downloads/details.aspx?FamilyID=3b882eb1-5f06-41d9-baba-d39ad13bc3ff&displaylang=en/) (2002).

**Category 3:** Issues that appear to have no obvious cause and therefore no obvious solution

**Category 4:** Issues that may be caused by other factors (e.g., discussion guide, interaction with participant, etc.)

Ideally, the RITE user researcher will have experience in the domain and in the problems typically experienced in this domain. If the researcher doesn't have this experience, it may be difficult for him or her to determine if an issue is likely to be a problem for other users.

Second, the product's decision makers should make time to observe user sessions and contribute to the design changes. Without their involvement, it is difficult to evolve the design, which is the essence of RITE. Last, the team should be able to rapidly interpret the results and make design changes, another defining attribute of RITE.

I've used RITE for a variety of software platforms—desktop, web, iPhone—and found it much more efficient than “traditional” usability testing. For example, when testing one app, we discovered a line of text that prevented participants from moving beyond the welcome screen. After we addressed this issue, we were able to press ahead, uncovering more critical conceptual issues. However, RITE does have its limitations. Category 2 issues can be difficult to address within the study timeline (e.g., the solution may not be clear), requiring several days to brainstorm, sketch, and refine the app. To alleviate this problem, try to allocate extra time between sessions, knowing that sometimes even a few days is not enough.

## PAPER PROTOTYPE TESTING

Before discussing paper prototype testing,<sup>5</sup> I'd like to address any potential confusion. Paper prototyping falls into a fuzzy area, given that it's a form of prototyping (see Chapter 7, “Prototyping App Concepts”) and arguably also a usability-testing method. While it has similarities to “traditional” usability studies and RITE, there are many notable differences. First, paper prototype studies typically include three roles: the moderator, the “computer,” and the note taker/videographer. The “computer” is the person who swaps out different screens and user interface controls depending on the research participant's actions. Second, paper prototyping usually includes some level of participatory design, where the user actively contributes to the design. For example, the participant might use a pen or pencil to rename items in a tab bar, or rearrange a layout. As we'll discuss later in the chapter, in these cases you should have paper prototyping materials on hand: Post-its, pens, glue, and so on.

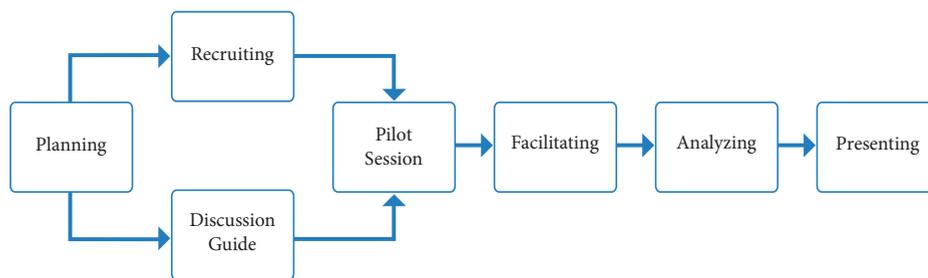
### NOTE

Having a three-person paper prototype team may be ideal, but it's certainly not required. I've conducted paper prototype studies where I played all three roles—it's doable but exhausting. If you can pull together a two-person team, I suggest that one person play moderator/computer and the other take notes.

5. Carolyn Snyder, *Paper Prototyping* (Morgan Kaufmann, 2003).

## Usability-Testing Timeline

If your prototype is ready and your recruiting requirements are relatively straightforward, you may be able to complete a usability study in less than two weeks. Other factors that may affect your timeline include the usability method and the study context. For example, iterative studies need design time between sessions, and field-based studies require more overall time (extra travel time and slightly longer sessions). **FIGURE 8.2** illustrates the key activities included in most usability studies. The activities occur serially, with the exception of recruiting and discussion guide creation, which often happen in parallel.

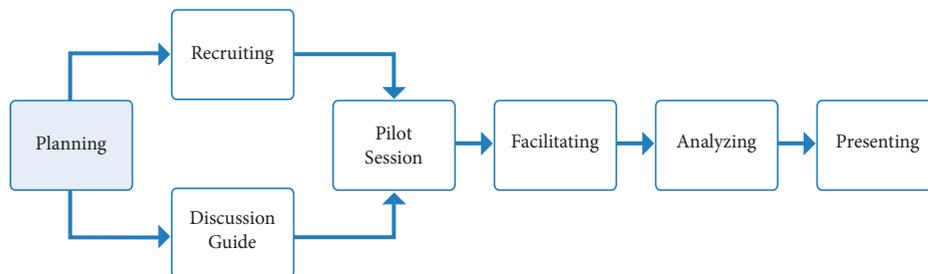


**FIGURE 8.2** Usability study activities

### NOTE

The high-level activities in **FIGURE 8.2** are essentially the same ones discussed in Chapter 3, “Introduction to User Research,” but the execution is different. In particular, there are major distinctions when it comes to the discussion guide and facilitating. These are discussed in the following sections.

## Planning Usability Tests



Usability planning often begins with a kickoff meeting with the app’s stakeholders—designers, developers, researchers, product managers, customer support. If your team has not participated in usability studies in the past, spend some time explaining the benefits and methods outlined earlier in this chapter. Topics discussed in the kickoff meeting and documented in the test plan typically include<sup>6</sup>

- Purpose and objective
- Study dates and times

6. Jeffrey Rubin, *Handbook of Usability Testing* (Wiley, 1994).

- User profile
- Method
- Questions for research
- Roles (moderator, observers, documenter)
- Prototype supplies (if paper prototype)
- Equipment and location
- Report contents

## PURPOSE AND OBJECTIVE

Articulating the study purpose and objective will help keep your study focused and ensure that everyone on your team is on the same page. This section includes three examples at different stages in the design process; the app cited helps users find local art events.

### Example 1: Early-Stage Research

- **Purpose**  
Evaluate the flows and interaction included in the current app design, with emphasis on finding events and getting directions to venues.
- **Objective**  
Uncover user experience issues and improve designs before development begins.

### Example 2: Baseline Before Development Begins

- **Purpose**  
Evaluate the overall user experience of the app design, including flows, interaction design, and lower-level details such as transitions.
- **Objective**  
Uncover user experience issues and improve designs before launch.

### Example 3: Feature- or Flow-Specific Research

- **Purpose**  
Evaluate the “share via Twitter” flow.
- **Objective**  
Uncover user experience issues and improve Twitter sharing before launch.

## STUDY DATES AND TIMES

Communicating the study dates and times will enable team members to block off their schedules so they can observe sessions. If you're testing "live" code, be sure to communicate the schedule to your development team. Without knowing your plans, they might make changes that could disrupt the study. Ideally, some members of the development team should also observe sessions.

When scheduling the study times, make sure you allocate enough time between sessions. If you're working with paper, you may need some extra time to "reset" the prototype between participants. "Resetting" a paper prototype may involve tasks such as rearranging the screens or erasing content handwritten on screens. Higher-fidelity prototypes may also need to be reset if participants added or removed content.

## USER PROFILE

The user profile, discussed in Chapter 3, may also be used for usability testing.

## METHOD

Your usability plan should specify which methods you plan to use and define them as needed. If you plan to gather usability metrics (e.g., task completion times and severity ratings), include them in this section as well as the method for gathering them (manual or automated).

## QUESTIONS FOR RESEARCH

Questions for research can cover everything from high-level conceptual issues to low-level app interactions. For example, when conducting a paper prototype study for the art events app, my research questions were designed to shape the app direction and included the following:

- Are prospective users interested in the app concept?
- Does the high-level feature set meet their needs?
- Are they able to navigate between tabs and screens?
- What additional content, if any, do they need on the event detail view?
- What are their impressions of the ad placements?
- Based on what they see that day, would they download the app?

## ROLES

The study roles will be influenced by your goals, context, and resources but may include the moderator, the note taker, the videographer, and the "computer" if

you're conducting a paper prototype study. As discussed in Chapter 3, your team's user researcher should play the role of moderator. If you don't have a dedicated user researcher, choose a moderator with the following qualities: patience, empathy, flexibility, and assertiveness. If no one on your team meets these criteria, you may want to outsource the moderator role.

When observing users out in the field, keep in mind that the environment will influence the roles. For example, if "the field" means observing a participant using your app in an office, it may be easy to bring one or two team members along. However, if "the field" means following a participant during the morning commute, you may be more nimble on your own.

If team members are observing your study, make sure you explain the observer role. One of the biggest problems I've encountered is observers frequently interrupting user sessions with questions. Observer questions are fine, but it's best to wait until a natural break in the script or the end of the session. Constant interruptions may confuse participants and distract them from the task at hand. Communicating these issues in advance will allow your study to run more smoothly.

## PROTOTYPE SUPPLIES

If you're testing a paper prototype and users may contribute to the design, be sure to have supplies on hand. The supplies should be similar to the materials used to create your prototype (e.g., Post-its, pens, extra paper, glue, etc.).

## EQUIPMENT AND LOCATION

When conducting studies in a lab, most researchers take notes during the session, supplementing them with audio and video recordings as needed. **FIGURE 8.3** shows a lab setup with two video cameras: one on the iPhone and hands, the other on the participant's face. **FIGURE 8.4** shows the observation room for the same study.

In contrast, if you're conducting your research in the field and the user is constantly on the go, it may be challenging to take notes as you observe. In this case you may want to capture the user's screen with a small mounted camera and record comments through a microphone. More complex setups can include additional cameras as well as wireless transceivers that let the moderator view the participant's screen from a distance.<sup>7</sup> Additional tips on mobile usability-testing configurations can be found on the Little Springs Design web site.<sup>8</sup>

---

7. Antti Oulasvirta and Tuomo Nyysönen, "Flexible Hardware Configurations for Studying Mobile Usability," *Journal of Usability Studies* (February 2009), [www.usabilityprofessionals.org/upa\\_publications/jus/2009february/oulasvirta1.html](http://www.usabilityprofessionals.org/upa_publications/jus/2009february/oulasvirta1.html).

8. Little Springs Design, [www.littlespringsdesign.com/](http://www.littlespringsdesign.com/).



**FIGURE 8.3** iPhone app usability study, with moderator and participant. The moderator (left) can see the iPhone projected on the large monitor.

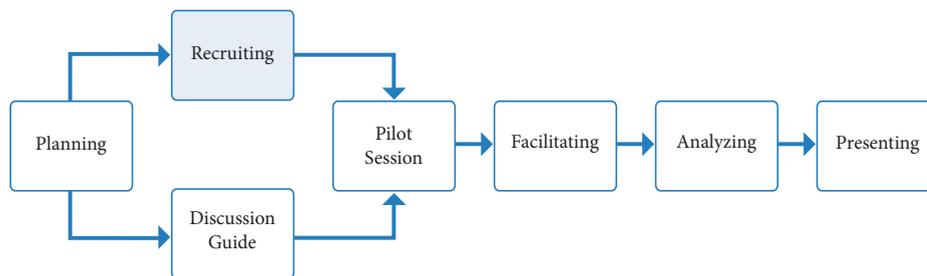


**FIGURE 8.4** iPhone app usability study observation room. The large screen (left) projects the iPhone screen while the participant is working through tasks; the small screen (right) shows the participant's face. Audio is also streamed into the observation room.

## REPORT CONTENTS

Given the rapid pace of iPhone development, most companies seem to prefer lightweight usability reports (e.g., an executive summary with your top findings and recommendations). However, every company is different, so it's a good idea to discuss your reporting strategy in advance. If key members of your team are unable to observe the sessions, you may want to include video and/or audio clips in your report.

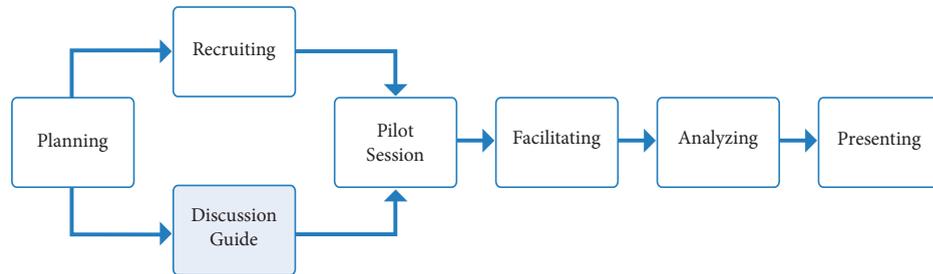
## Recruiting Participants



In Chapter 3 we explained how to create a participant screener and discussed several recruiting options—friends and family, recruiting agencies, Craigslist—which are also appropriate for usability testing. As mentioned earlier, don't

underestimate the importance of recruiting individuals who match your user profile. Five to 8 participants is generally recommended, but recruit 12 participants to account for no-shows and a pilot session.<sup>9</sup>

## Drafting the Discussion Guide



After the recruiting process is under way, start drafting the discussion guide for your study. Discussion guides often differ based on the study and the practitioner style. I’ve found the following format to be effective for a 90-minute study; there are 15 extra “floating minutes” for getting situated and possible bathroom breaks. Do not exceed 90 minutes, as participants may get tired and lose patience. It’s also a good idea to provide water or other beverages.

- Introduction (5 minutes)
- Background interview (15 minutes)
- Tasks (40 minutes)
- Follow-up questions (10 minutes)
- Wrap-up (5 minutes)

### INTRODUCTION (5 MINUTES)

Provide your name, your company, and information on the process. In particular, explain “thinking out loud” and other method-related information. For example, you might say, “Please describe what you are doing. Imagine that you are talking to a friend who can’t see what you’re doing.” If the app is currently in paper form, it’s important to explain how to interact with the prototype (e.g., how to select items and how to enter text). If an NDA (a nondisclosure agreement) or other documents are required, ask the participant to sign them before you begin the study. Finally, inform the participant if team members are observing behind a one-way mirror.

#### TIP

A sample NDA can be found at the Society for Technical Communication’s web site, as part of its “Usability Toolkit” ([www.stcsig.org/usability/resources/toolkit/toolkit.html](http://www.stcsig.org/usability/resources/toolkit/toolkit.html)).

<sup>9</sup> Jakob Nielsen and Thomas K. Landauer, “A Mathematical Model of the Finding of Usability Problems,” *Proceedings of ACM INTERCHI '93 Conference* (Amsterdam, April 24–29, 1993), 206–13.

## BACKGROUND INTERVIEW (15 MINUTES)

Confirm responses from the participant screener and probe deeper as needed. These interviews are a good opportunity to ask participants to *show* how they use related products. For example, when I screened participants over the phone, we discussed the iPhone apps they were using for local event information. When we met in person, I took this one step further and asked them to demonstrate how they used these apps. Try to limit background interviews to 15 to 20 minutes.

## TASKS (40 MINUTES)

The number of tasks will vary depending on the app and the estimated task duration. You may have four tasks that take approximately ten minutes each to complete, eight small tasks that require five minutes each, and so on. Whatever the breakdown, try to start with an easy task and provide a natural flow. For example, if you were testing a photo-sharing app, the natural flow might be image capture, image editing, and then image sharing. All of the tasks should relate to the objective outlined in your study plan.

## FOLLOW-UP QUESTIONS (10 MINUTES)

Follow-up questions provide an opportunity to step back and understand the participant's impression of your app. Questions that often elicit insightful answers include the following:

- What is your overall impression of what you saw today?
- Do you have any concerns?
- Let's say you wanted to describe the app to a friend; what would you say?
- Is there anything else you wish it could do?
- Would you buy/use the app?

## WRAP-UP (5 MINUTES)

Thank participants for their time and contributions to your app. Ask them if they have any outstanding questions, then provide the necessary payment.

**FIGURE 8.5** contains an excerpt from a discussion guide created for the previously mentioned art events app. The primary goal of the study was to understand whether prospective users were interested in the concept and whether the initial design met their needs. The prototype medium was paper (**FIGURE 8.6**), and the sessions were conducted in a conference room in San Francisco, California. Participants were recruited using Craigslist and an online survey. The 90-minute study included three tasks; only one is shown in the figure.

## INTRODUCTION

“Thanks for taking the time to meet with me. My company is developing a new product, and it helps if we learn more about people’s unique experiences with the technology. First, there aren’t any right or wrong answers, so don’t worry about giving us ‘good data’—it’s all interesting to us, no matter how boring you think it is.

“Today’s session will be divided into a couple of different sections but will take no longer than 1.5 hours. Before we begin, I need you to sign a nondisclosure agreement.”

## TASK 1. FIRST-TIME USER EXPERIENCE

“Imagine that your friend told you about a new iPhone app that helps you find local art. Let’s say that you download the app to your phone that weekend and decide to try it out. Have you ever been to Ritual Coffee in the Mission? Imagine that you start your day at the café. When you start the app, the following screen appears:” (*Show start screen.*)

Before the participant clicks on anything, possible probes:

- Is this what you expected to see?
- What did you expect? (*if the participant did not expect this content*)
- What would you do next?

## UI QUESTIONS

If the participant does not comment, probe into the following before proceeding:

- What do you think about the event information shown? Is anything missing or unclear?
- What do you expect to happen when you click on Sort? (*Note the sort values expected.*)
- What do you expect each of these to do? (*Point to the bottom toolbar.*)

[*Tasks 2 and 3 appeared here.*]

## FOLLOW-UP QUESTIONS

In addition to the follow-up questions described in the previous section, participants were asked:

- Which galleries need to be included to make this useful to you?
- If the service required an account, what would make this most compelling?
- Would the app replace anything you already use?
- Would you be willing to pay for this app?
- How do you feel about ads on this app? What kind of ads do you think would be appropriate?

**FIGURE 8.5** Excerpt from the discussion guide for an art events iPhone app paper prototype study.

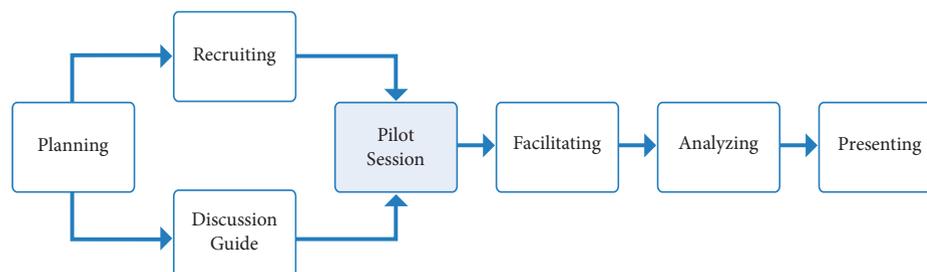


**FIGURE 8.6** iPhone paper prototype for finding local art events. An iPhone was used to record audio and photograph participants.

### We will need a heading for this new sidebar please

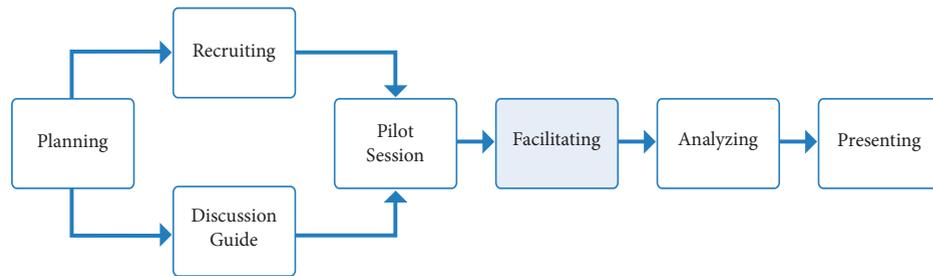
To test out your discussion guide, consider reading it out loud as you walk through the user interface. This activity will help you uncover flaws in your guide. For example, you may discover points where you may want the user to pause so you can ask a question. While this solo walk-through is tremendously valuable, it's not a substitute for a "pilot," which is essentially a dry run of the test with a target user. If everything goes smoothly, you may be able to incorporate this data into your findings. Otherwise, plan to revise the prototype and/or guide and exclude the data.

## Pilot Session



As mentioned earlier, it's critical to run a pilot session to uncover any issues with the prototype or discussion guide. In addition to recruiting a participant based on your user profile, you may want to do a test run with a team member unfamiliar with the app design. Be sure to run the pilot at least a few days before the actual study; that way you'll have enough time to make revisions.

## Facilitating Usability Tests



The discussion guide will be instrumental as you facilitate the usability-testing sessions. As mentioned previously, remember it's a “guide”; thus you should adapt as needed. At the same time it's important to know when to draw participants back into the guide. For example, the art events app prototype did not include a companion web site. However, when a participant asked about the web site, it seemed worth exploring, so we discussed this topic further.

### NOTE

Tasks can be provided orally, written on paper, or both. I tend to provide both.

In contrast, when another participant started to discuss a feature that was clearly out of scope, I noted the comments and politely moved on to the next task. Additional facilitating tips are discussed in the next section.

### BE ENCOURAGING

After presenting users with each task, remain quiet, giving them a chance to orient themselves. Resist any temptation to provide hints or explain what to do next. Provide encouraging verbal and nonverbal feedback such as nodding and smiling.

### ASK OPEN-ENDED QUESTIONS

As participants are working through tasks, you may ask open-ended questions to better understand their behavior and comments. Open-ended questions are also an effective way to help participants when they seem confused. Here's an example of this type of dialogue:

**Participant:** *(Quietly staring at screen)*

**Moderator:** “What are you thinking?”

**Participant:** “Well, I was going to tap here, but I’m not sure if it will show more event info.”

**Moderator:** “What do you expect it to do?”

**Participant:** “I’m not sure. I think it will provide more info on the venue and opening times.”

**Moderator:** “What would you do if you were at home [*or wherever the app is used*]?”

**Participant:** “I would probably try it and see what happens.”

**Moderator:** “Okay, let’s try that.”

**Participant:** (*Taps on selection*)

**Moderator:** “Is this what you expected?”

**Participant:** “Yes.”

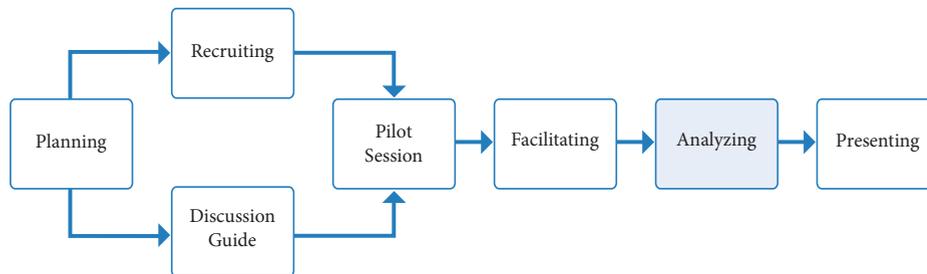
#### NOTE

If participants try to tap on a feature that has not been defined yet, it’s a good idea to ask what they expect the feature to do, then tell them it’s not working yet. Knowing their expectations may help shape the feature requirements and design solution.

## KNOW WHEN TO STOP

If the participant has been trying to complete a task for a while and seems aggravated, allow some recovery time, then move on to the next task. Sometimes I’ll explain what happened and answer task-related questions at the end of the session. When I explain how we intended the app to work, I always add, “Thank you for your help in identifying that problem—your feedback will make the product better for everyone.” This reassures participants that *they* weren’t the problem; the *interface* was the problem. This exchange may also lead to additional insights.

## Analyzing Usability Tests



In contrast to up-front user research analysis, which may involve several team members, usability tests are typically analyzed by the user researcher. One reason is objectivity—the app designers may be too attached to their designs; the other is speed. Having your entire team collaboratively analyze each task will take much

longer. This doesn't mean that the researcher does not consult with the designers and developers; it just means that the researcher leads the analysis effort. Of course, this assumes you have a dedicated researcher. I've played the role of designer and researcher and have been able to objectively analyze the data, but not everyone can.

Regardless of who leads the effort, consider creating affinity diagrams of your observations, as discussed in Chapter 4, "Analyzing User Research," using your notes as a starting point. Groupings will vary for each app—they can be organized by task, themes, severity, scope. For example, for the art events app, I organized my findings into overall issues and task-specific issues. Similar to the approach in Chapter 4, I tend to summarize the findings in one to two sentences, then provide supporting quotes. If you captured audio or video, you can create short clips to accompany these findings. A few excerpts are shown in the next section.

#### NOTE

*P* followed by a number is shorthand used to refer to participants from the study. The number indicates when the participant was interviewed; for example, P1 is the first participant.

### OVERALL ISSUES

Participants expected the app to have a companion web site.

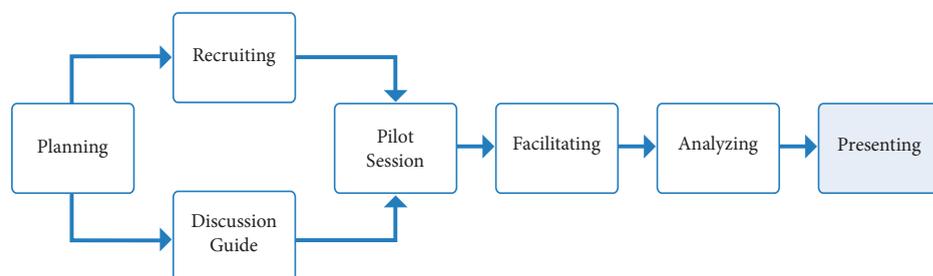
- "I'd rather use the web site when I'm at home since the screen is bigger." (P3)
- "Can I write reviews through the web site?" (P4)
- "Would the bookmarks be the same on the web site?" (P5)

### TASK-SPECIFIC ISSUES

Participants expected additional event information.

- "It'd be nice to see the material and dimensions for the artwork shown." (P1)
- "When are the opening and closing receptions?" (P2)
- "Where is the telephone number? I like to call to make sure they're open." (P5)

## Presenting Usability Findings



After analyzing your user sessions, consider writing up a Quick Findings report at a minimum. As discussed earlier, Quick Findings typically include approximately five to ten of the most critical issues, sometimes called “top findings,” followed by more detailed findings. The extra effort may seem unnecessary at the time, but the document will be a valuable reference in the future.

Although your observations are fresh immediately after the study, it will be hard to remember the details a few weeks later. Once you have your findings documented, share them with the stakeholders who attended the original kickoff meeting.

In addition to sharing insights, you may want to use this meeting time to start brainstorming and prioritizing solutions. Be sure to discuss next steps while the study is fresh in everyone’s mind. Unfortunately, as time passes, many teams start to minimize what seemed like critical issues.

## Guerrilla Usability Testing

Guerrilla usability testing is derived from *guerrilla marketing*, a term that was coined by Jay Conrad Levinson in his book *Guerrilla Marketing*.<sup>10</sup> Like its marketing counterpart, guerrilla usability testing relies on time, energy, and imagination instead of a big budget. Given its unconventional nature, there aren’t any set rules or definitions. In fact, when I was researching user experience organizations that provide guerrilla testing, I found it hard to distinguish their methods from “traditional” usability testing. With that in mind, be aware that practitioners may have different opinions on what constitutes guerrilla usability testing. The next section introduces a few variations that are relatively popular.

### COFFEE SHOP TESTING

“Coffee shop” researchers typically go to their local coffee haunt in search of study participants (Starbucks seems to be quite popular). Once they identify an acceptable candidate, they offer the individual coffee in exchange for evaluating their app. These sessions tend to take an exploratory approach; for example, a developer might hand the participant an iPhone with the app and ask, “What would you do when you first open this app?” The location doesn’t necessarily have to be a coffee shop. The creator of the What’s Shakin’ app used a similar strategy at a bar in San Francisco.

---

10. Jay Conrad Levinson, *Guerrilla Marketing* (Mariner Books, 2007).

## WALK-UP TESTING

“Walk-up” testing is similar to “coffee shop” testing in the sense that you approach strangers; the main difference is context and duration. “Walk-up” testing usually takes place on the sidewalk or other public place. Given that these people are typically en route somewhere, they may have less patience than coffee shop customers. Ideally, you should provide some type of modest incentive (e.g., a coupon code for your app or an iTunes gift certificate).

## COMMON GROUND TESTING

You may have more leeway if you can conduct guerrilla tests in a place where you have something in common with the individuals. For example, when attending a conference for wine professionals, Hello Vino representatives asked fellow attendees to try out their iPhone app prototype. In exchange, they gave each participant a free wine charm. Depending on your app, there may be other venues that are suitable for guerrilla usability testing, but please choose the location and approach wisely. Holding interviews in certain locations may be unacceptable for legal or privacy reasons.

### Word of Caution

When I suggest up-front usability testing to iPhone developers—that is, “traditional” usability testing—they typically nod their heads but don’t seem convinced. However, when I suggest guerrilla methods such as the ones described here, their eyes light up as if to say, “Sure, I can handle that!” I suspect their change in attitude is largely because these nontraditional alternatives seem fast and easy, whereas the other approaches require more planning.

Given that some usability testing is better than no usability testing—not always, but usually—I urge them to at least try one of the guerrilla approaches. But, at the same time, I explain what they’re missing by forgoing the traditional usability testing or RITE route.

## Representative Users

Without proper screening, it’s unlikely that coffee shop or walk-up participants will adequately represent your user base. And, unless they’re holding an iPhone, you may not even know if they’re iPhone users. Sure, you can stop them on the street and ask, but if you get several “no” answers, you might have saved time by simply recruiting up front. Last, this hit-or-miss approach is clearly a deal breaker if your app is designed for a specialized group, such as doctors.

## Adequate Time for Tasks

Nearly all of the guerrilla approaches mentioned imply that someone is doing you a *quick* favor. As a result, chances are you'll keep the study brief, limiting participants' exposure to your app. This might be fine if your app is a game or utility that's typically used for very brief periods of time. However, if you've designed a productivity app with a rich set of features, the value of a guerrilla user test may be minimal. In this case a guerrilla test may be useful for gathering initial impressions but inadequate for uncovering deeper user experience issues.

## Ethics and Supporting Documentation

As discussed in the previous section, you should always begin usability studies with an introduction of the procedure, emphasizing that you're testing the app, not the people. Additionally, in many cases companies ask participants to sign an NDA and release forms for using the research and/or photos. You can certainly include these documents when conducting guerrilla tests, but the impromptu nature often leads individuals to forgo these important steps. You may not run into any problems, but do you want to take a chance?

## Beta Testing

iPhone beta testing is made possible through Apple's Ad Hoc Distribution system.<sup>11</sup> Based on a series of phone interviews with developers, it's my understanding that most of them find participants through their personal networks, through forums on their web sites, or through social networking services like Twitter. The feedback tends to be unstructured, and the level of detail varies from one participant to the next. This is unfortunate, because beta testing can be very powerful—participants' feedback can be gathered in situ, in context; the participant pool can be geographically dispersed; and feedback can be collected over an extended period of time. To make the most of beta testing, consider enhancing it with some user-centered techniques such as the following ones.

### CAST A WIDER RECRUITING NET

The Ad Hoc Distribution system enables developers to include up to 100 participants. Instead of limiting yourself to friends and loyal Twitter followers, create a user profile, as discussed in Chapter 3, and recruit participants who match the profile.

---

11. iPhone Developer Program, <http://developer.apple.com/iphone/program/distribute.html>.

## ASK FOR MORE STRUCTURED FEEDBACK

Include a survey with your beta and consider associating survey questions with specific app tasks. For example, if your beta includes a new email-sharing feature, you could display a brief survey immediately after the user has sent a message. Alternative approaches for prompting users—random, scheduled, event-based—are discussed in detail in the paper “Using the Experience Sampling Method to Evaluate Ubicomp Applications.”<sup>12</sup> In terms of tools, there are a number of options for collecting iPhone user data. Haveasec.com can help you gather qualitative data via surveys, and services like Flurry, Pinch Media, and Mobclix use analytics to collect quantitative data (e.g., time on task and drop-off rates).

## PROVIDE AN INCENTIVE

Many developers are disappointed when testers fail to follow through on the beta. One way to improve response rates is to provide an incentive after participants submit feedback (e.g., an iTunes gift certificate).

## Choosing an Approach

Determining your usability-testing strategy will depend on a variety of factors—the study goals, the app, your team’s skill set, the available time and budget. At a minimum, I recommend two studies: one in the early design stage and another in the later stages. If paper is a suitable medium for prototyping your app, consider a paper prototype study for the initial test and a device-based RITE or “traditional” study for the later test.

On the other hand, if paper is not effective for your domain (e.g., musical instruments), you may want to test your app on the device early on and conduct a usability test with your beta in the final design phase. Finally, if your time and budget are extremely limited, consider one of the guerrilla usability-testing methods discussed in this chapter.

## Summary

This chapter discussed the benefits of usability testing. In addition to impacting the bottom line—fewer costly iterations on “live” code—usability testing can lead to increased customer satisfaction. Instead of waiting until your users uncover

---

12. Sunny Consolvo and Miriam Walker, “Using the Experience Sampling Method to Evaluate Ubicomp Applications,” *Pervasive Computing* (April–June 2003).

issues and vent on the App Store, you can address these issues before even coding your application.

There are many different usability-testing approaches. You were introduced to “traditional” usability studies, the RITE method, paper prototyping studies, guerrilla testing, and beta testing. While each app is different, most apps can benefit from at least two user tests: one in the early design stage, and another one in the middle to later design stages.

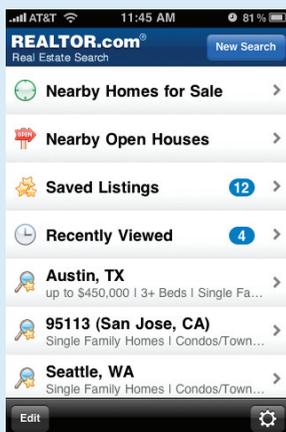
- Some research is better than no research. If you can’t run a RITE or traditional usability study, try a guerrilla method or create your own!
- Be as inclusive as possible when planning and running your usability tests. Having your team members on board will make it easier to integrate findings into your app.
- Make sure you pilot your usability study *at least* a few days in advance. This will give you time to adjust the discussion guide and prototype. ■



## CASE STUDY

# REALTOR.com

**CLIFF WILLIAMS** has been designing and building user experiences for the web and desktop for the last 15 years. He's currently the interaction design lead for mobile user experiences at Move, Inc., operator of Move.com, Moving.com, SeniorHousingNet.com, Top Producer, and REALTOR.com, the number-one site for home sales.



**FIGURE CS8.1** REALTOR.com home screen

### What inspired REALTOR.com to build an iPhone app?

A big part of searching for a home is being out and about—checking out neighborhoods, going to open houses, touring with an agent. A REALTOR.com iPhone app felt like a natural and useful extension of our web experience.

One of our developers created the spark with a proof-of-concept app. That attracted me and a couple of others at the company. From there we spent whatever extra cycles we could find—between projects, nights, weekends—refining the concept. Eventually it developed enough interest and became an official project (great apps from our competitors helped, too).

### How did you approach the project?

Early on it was a mix of open brainstorming and refining the proof-of-concept app. This was a great approach for us because it let big new ideas flow in while we were learning the bounds of the API and the capabilities of the platform.

Once it became an official project, we took a step back to evaluate where we were with the proof-of-concept app, the pool of ideas we had developed, and past user research that might be applicable. This helped us really home in on a feature set focused on the mobile user.

### What were some of the challenges you faced?

As an experienced web designer but first-time app designer, it took a while for me to really get comfortable applying the patterns and principles of the platform. My first attempts were essentially 320 × 480 web sites, not iPhone apps. Data was displayed in long sections that required lots of scrolling. The architecture focused on presenting a breadth of choices at the expense of a crisp, simple workflow. As the app evolved, it became more and more iPhone-y.

### Were you able to conduct usability tests?

After our first big round of design on the official app, we started thinking about usability testing. Getting a live prototype in time wasn't possible, so I started experimenting with simple HTML prototypes that we could test on a working iPhone. These were effective, but it was difficult to simulate scrolling with fixed-position elements.

We ended up going with a paper prototype that included all of the screens in the app, variations of some key screens, and a handful of bits and pieces that we would overlay at different points. These were created with Adobe Fireworks at 72DPI. Next time I would create them at 300DPI since the printouts weren't that sharp.

Testing netted one big issue (saving your favorite items was confusing) and several smaller ones (clarity of icons and copy). These results pushed us even further down the reduce-simplify-streamline path our designs had been evolving along. [FIGURES CS8.1–CS8.3 show the final designs.]

### Can you describe the usability-testing environment?

Evonne Shea, our project's user researcher, started each session with a brief interview to gather high-level information about the user's current real estate search needs and experiences. Afterward she'd introduce me (the computer) and then move to the adjacent observation room. From there she could observe the participant through a one-way mirror as well as a pair of video cameras—one focused on the prototype, the other on the participant's face. Video was recorded for later analysis and also displayed on a projector in the observation room so team members could watch. We included eight participants in the study. All of them were currently involved in searching for real estate and had some experience downloading iPhone apps.

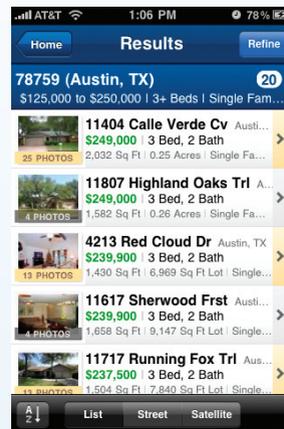


FIGURE CS8.2 REALTOR.com search results



FIGURE CS8.3 REALTOR.com map view

### Any additional advice for iPhone app designers?

If you're not developing the app yourself, learn to speak the same language as the folks who are. Read Apple's *Human Interface Guidelines* to start. It will help get the ball rolling on your designs, and it's invaluable as a Rosetta Stone. Check out the online API documentation, too. For purposes of specifications, it's critical to know what the API provides "for free" and what it doesn't.

Another thing that helped me a ton was creating a giant library of screenshots. Download every app that you hear good things about and take screen captures of anything you see that is remotely interesting. Perhaps even more important, take screen captures of the default Apple apps (even the boring stuff, like Settings, is a gold mine). I constantly referred back to this library for best practices on some of the more mundane things and inspiration for big new ones. ■